Vernier Software
Jumpstarting the STEM revolution

By Kay Savetz

First published in Juiced.GS Magazine, December 2019

Dave Vernier is the co-founder, with his wife Christine, of Vernier Software & Technology, a company that sells software and hardware for science education. The company's first products were for the Apple II computer. Vernier is author of How to Build a Better Mousetrap and 13 other Apple II Projects, published in 1984. He still heads the 117-employee company, based in Beaverton, Oregon. I visited Vernier at the company headquarters, where he gave me a tour of the facility. We sat in a room he called the computer museum, filled with all manner of 8-bit computers, calculators, and educational electronics, for this interview.

Kay Savetz: Before you were a programmer and publisher, you were an educator. How did that career introduce you to computers?

Dave Vernier: I graduated [from Ohio State University] in 1969 and started teaching in inner-city Cleveland. Then in 1973, I came out to Oregon State, got a master's degree, and taught in Hillsboro [Oregon], which was a good suburban school in those days. [Scientific measuring device company] Tektronix was a huge part of the Oregon economy. Schools were decently funded.

I started fooling around with the terminals and writing little programs for my classes. I took one programming class in college; it used punch cards, and it was horrible. But I saw right away that computers could do things to help my physics teaching. Even with the punch cards, I could do that. Then I was able to get… I think the first computer I got was a [Exidy] Sorcerer, which doesn't exist anymore, but it was a real simple little thing. And I could start writing simulations on it and create graphing programs and grading programs. Then Tektronix gave us a graphics terminal, so you could write pretty nice graphics programs. I did things like simulating satellite launches and projectile motion. I spent all my time at work. My wife's yelling at me constantly because I'm not coming home, and she's got dinner ready and stuff.

I was able finally to get an Apple II of my own, and then I could do all this programming at home. I started writing these programs just for my own use. There was no illusion about starting a company. The first ones were some simulations of things that we couldn't do in a lab. I was a total believer in hands on education, but some things you can't do like launch satellites or launch projectiles at high enough speeds—air resistance becomes a huge deal. Just letting you do some different things, changing gravity, and stuff like that.

I was totally into doing labs, and the kids were good at doing labs, and they seemed to really enjoy the labs. But the trouble was they'd take data, write it down on paper, and then they'd have to make a graph. That act of making a graph was tedious, and they weren't good at it. By the time they had made the graph, the connection between what they did earlier, 15–20 minutes ago, and that piece of paper, was totally lost. To them, the graph was a busy-work exercise, sort of like connecting the dots. So, if we could find a way to have a graph appear almost instantly, then you could start looking at, what's that graph mean?

That huge breakthrough led to the program Graphical Analysis, which we wrote in 1981. We still—38 years later—have a version of that program, because getting a graph made quickly and efficiently is important.

It was very obvious that computers helped me to be a physics teacher. In 1981, the Oregon economy was terrible. There weren't any summer jobs: my wife's going to work at six in the morning, and I'm home and there's no job. I said, "Well, hell with it.

I'll spend the summer working on these programs for my class and, worst-case scenario, we'll have better programs for my class. And maybe we'll even try to sell them." That's how the company started.

I would continue on the weekends to work on programs, maybe add one or two new programs a year and improve things, fix bugs, and stuff like that. And after a year or two we said, "We're actually selling a little bit." My wife volunteered to quit her job and stay home and be there, and she could do the business stuff and get orders ready. That went on for about a year. We were still selling stuff and maybe growing a little, but it was not quite working because I wasn't there to write programs, and so we were not developing a lot of new products. We decided, well, I better stay home. She had to go—she worked at Nike, actually, to pay the rent, and I took a leave of absence from school. Then I was full-time and could crank out some programs quicker. We did that for, I think, two years and then she was able to quit. We had two full-time people. And we've grown from that to about 110 right now.

KS: What were your first programs?

DV: We had the graphing program and some simulations of orbiting satellites, charged particles moving in electric fields, projectile motion. Wave superposition and vector addition. Kinematics was actually interesting. It was a sort of a simulation: there was a little toy truck that would move across the screen. Really low resolution, horrible graphics, and it would leave a dot every second. You would program this truck to either accelerate or start at such-and-such speed and accelerate with this such-and-such acceleration. The idea was, how did those dots space out? Sort of like a ticker-tape timer. (That's what physics teachers called the classic lab where you'd get marks.)

KS: Do you recall why you started with the Apple II?

DV: I also dabbled in the Commodore 64. In those days, those were the two dominant computers for education. Remember, we're all about education. I didn't give a hoot about anything else. The Commodore for a while looked to me like it was a better deal and had more bang for the buck, but it just didn't survive. We wrote only one program for it. So the Apple II seemed to be the way to go. And it was.

The other thing that's slightly interesting is, remember we're selling this stuff to physics teachers, including physics professors at college. And one of the things we quickly discovered is that the physics professors thought of this Apple II as a toy. When IBM came out with the IBM PC, [professors said], "Whoa, that's a man's computer. That's a real computer." And they wanted that desperately. They were practically begging me to write the same programs over for the IBM. Finally I did. Actually got students to help me mostly with that. That was a big deal because for some reason college professors perceived the Apple II as a toy. IBM had that clout and the big name.

KS: The Apple programs were written in Applesoft BASIC?

DV: Yes. I listed the entire basic program in the back of the manual and had documentation: a list of variables, an explanation of the structure, and a memory map . The exception to BASIC was certain things where you needed speed! I would write the machine language loops. Like for example, the orbiting satellites thing. The math of actually moving the dot on the screen was done in machine language. I actually learned machine language, and it was a different world.

I was totally alone, not a single soul to talk to. I don't think I had one human being on Earth that I ever talked to about Apple II programming. There was nobody, and there was no Internet. I would buy books and read the book and then just rip into it. It is so different now. We have about twenty programmers now—it's a very, very social thing

where they get together. They seem to me to be in meetings more than they are programming, and it's quite interesting, and it would make the job a lot more fun I think.

KS: Do you program anymore?

DV: Yes, I do. I was programming when you arrived. Right now, I'm doing a micro Python thing on a Lego Mindstorms EV3, but this morning I did Raspberry Pi stuff and Arduino stuff. So, I'm always fooling around. It's low level, and it's just at the level of trying to get kids to learn programming. I bet I've done at least ten different platforms: Arduino, Raspberry Pi, Lego, something called mBot—it just goes on and on.

KS: Tell me about the hardware interfaces that you could hook up to the Apples for lab experiments.

DV: I mentioned that the first programs were simulations in a graphing program and some mathematical things, and we sold those for $18 per site license for the school.

KS: For a site license? Seems like a very generous price.

DV: You could say generous and stupid. It didn't take long to realize, "Hey, we're not going to survive." There's no way there can be a company. But at the same time, I always say I was realizing very quickly that, boy, one of the neatest things you could do with an Apple II is make measurements and display readings on the screen. Because there was no way before.

Let's imagine you were a chemistry teacher, and you want to show an exothermic reaction when the temperature increases. How do you do that demonstration in front of the class? Well, you'd have the chemicals there; you'd have a thermometer there; and you'd say, "Johnny, come up and read the thermometer. Tell the class what the temperature is." The class, including the person reading the temperature, is bored out of their skull.

But what if I could have an Apple II just displaying a number on the screen in big digits? Or even better, make a graph. It was a game changer! So, that little simple thing there was just a huge deal.

The first thing we did with that was temperature probes. We did voltage probes using the voltage frequency converter in the Apple II game port. And photo gate timing. That was a big deal, timing in a physics lab. But you need hardware for that.

Our solution: we had, in the back of the manual, instructions telling people to go to Radio Shack and buy these parts, and a diagram of how to solder them together, and you've got your sensor. That's what we did for a long time. We didn't sell hardware.

Finally, I remember distinctly one guy calling and he said, "Mister, why don't you sell a bag of those parts for that thing you're asking us to build?" And I said, "Well, you can buy it as easy as I can." He said, "I live in Wyoming; it's 75 miles to the nearest Radio Shack." He said, "You idiot—you can go to Radio Shack and buy 10 of each of those parts and you get a 10% discount, and you can make a little money."

We agreed to do it. We sold lots of these parts kits, and that was a really big deal. We'd sell hundreds and hundreds of parts kits, and people would solder together the parts. That went on for years and years. Then similar calls started to come in, people grumbling, "Look, why don't you build these temperature probes? I don't have time to build them." I'd say, "I don't have time to build them, either!" But finally, we said we'd do it. We hired college kids, literally in their dorms, to put together the things.

That went on with very homemade looking kits for years maybe. We finally went sort of commercial and had real nice equipment, which we do now. Now we develop for five platforms: Windows, Mac, iOS, Android and Chromebook. Chromebooks are taking over the world of education. It's just a total revolution.

KS: What was the span of years that you sold Apple II products?

DV: We live in a world where there's so much difference between the rich and poor schools. I bet we sold stuff to Apple IIs at least into the early 2000s. Or at least we would support them and maybe give them what's left. I had a friend who just told me the other day that he still uses one of our simulations just because—partly as a gag now, but it still works.

KS: Did you do any work with Apple directly at any point?

DV: You know, over the years I was down there probably a couple of times. I went down there for some Apple II developers conferences, but never really worked closely with them. One of the great frustrations in this business is big companies. We get to know one person from this big company, and we really like them. They like us. That one or two people that we know really well need to get the blessing of the big company. And that doesn't happen sometimes. And so there's a lot of bluster and a lot of excitement, and then it kind of fizzles out.

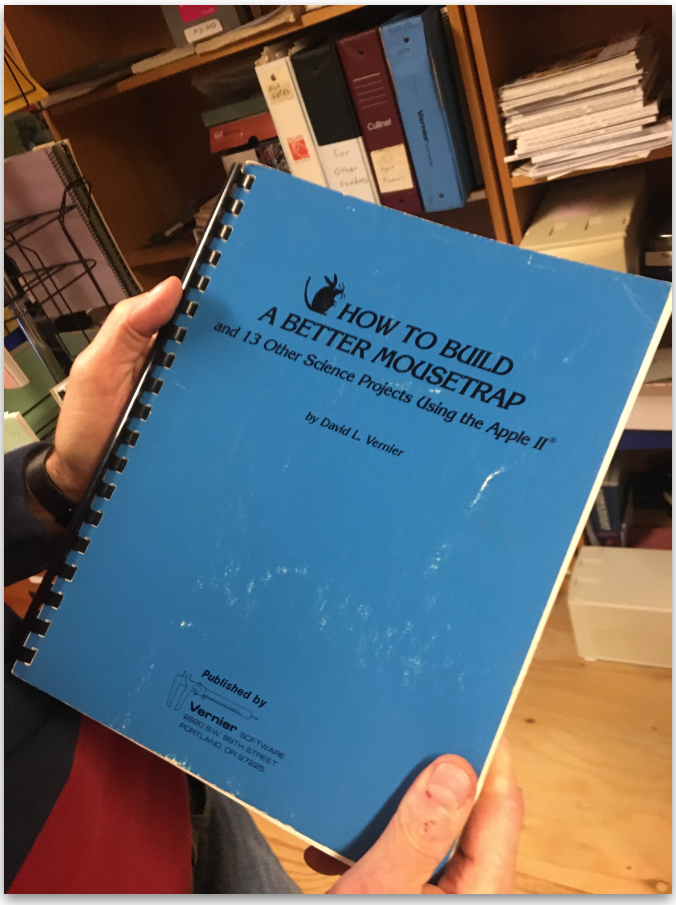
KS: Your target audience today is still education?

DV: Absolutely. 100%. Once in a while, we accidentally sell something to a non-teacher type, but we don't pursue it. And not for research, but for teaching labs.

KS: You said you have twenty or so programmers here—I assume they're young people. When they come into this room filled with old computers and see the Apple II machines, do they have any idea what's going on with them?

DV: I think they don't. Like I say, their world is so different from what mine was. Especially the nature of programming.

For the record, there's two Apple IIs here next to me that are running fine, all these years later: an Apple II Plus and an Apple IIGS. I am starting to have problems with some of the other computers. I think it's just plain corrosion. Like that IBM PC, the first one we've ever owned, quit working. I pulled all the chips out and jiggled them. That still hasn't solved the problem, so that's too bad.



KS: Your company helped start the revolution of “makers” and using computers for teaching science.

DV: We were STEM [science, technology, engineering, and mathematics education] before there was STEM. I wrote that book [How to Build a Better Mousetrap and 13 other Apple II Projects] in 1984, and I swear, it’s the same projects that they’re doing now.

Copyright 2019 Kay Savetz. This work is licensed under CC BY-NC-ND 4.0.